

REMARKS

In the Official Action mailed **May 2, 2002**, the Examiner reviewed claims 1-26. Claims 1-2, 5-10, 12-13, 16-17, and 20-25 were rejected under 35 U.S.C. §102(a) as being anticipated by Nakanishi et al (EPO 0 903 677 A2, hereinafter "Nakanishi"). Claims 3, 14, and 18 were rejected as being unpatentable over Nakanishi in view of Sudhakaran et al. (USPN 6,161,150, hereinafter "Sudhakaran"). Claims 4, 15, and 19 were rejected under 35 U.S.C. §103(a) as being unpatentable over Nakanishi in view of Ho (USPN 5,615,373, hereinafter "Ho"). Claims 11 and 26 were rejected under 35 U.S.C. §103(a) as being unpatentable over Nakanishi in view of Devarakonda et al. (EPO 0 655 495 A2, hereinafter "Devarakonda").

Rejections under 35 U.S.C. §102(a) and §103(a)

Dependent claims 11 and 26 were rejected as being unpatentable over Nakanishi in view of Devarakonda.

Applicant respectfully points out that Devarakonda teaches **distributed locking managers** (see Devarakonda, FIG. 1, and column 3, line 54 to column 4, line 10). In contrast, the instant application discloses **multiple independent locks** on lockable resources (see FIG. 2, and page 8, lines 8-19 of the instant application). Multiple independent locks are not the same as distributed locking managers. Having multiple independent locks is advantageous because it allows multiple controllers to lock independent portions of a lockable resource simultaneously. For example, referring to FIG. 2 of the instant application, controller 206 can acquire lock 228 on managed resource 112 while controller 208 can simultaneously acquire lock 229 on managed resource 112.

There is nothing within either Nakanishi or Devarakonda, either separately or in concert, which suggests that having multiple independent locks would be an advantage.

Accordingly, Applicant has amended independent claims 1, 12, and 16 to include the limitations of dependent claims 11 and 26. Dependent claims 11 and 26 have been cancelled without prejudice.

Hence, Applicant respectfully submits that independent claims 1, 12, and 16 as presently amended are in condition for allowance. Applicant also submits that claims 2-10, which depend upon claim 1, claims 13-15, which depend upon claim 12, and claims 17-25, which depend upon claim 16 are for the same reasons in condition for allowance and for reasons of the unique combinations recited in such claims.

Version with markings to show changes made:

The Claims:

1 1. (Once Amended) A method for providing concurrency control for a
2 policy-based management system that controls resources in a distributed
3 computing system, the method comprising:
4 receiving a request to perform an operation on a lockable resource from a
5 controller in the distributed computing system, wherein the lockable resource
6 presents one or more independent locks providing access to independent sub-units
7 of the resource;
8 wherein the controller sends the request in order to enforce a first policy
9 for controlling resources in the distributed computing system;
10 determining whether the controller holds a lock on the lockable resource;
11 allowing the controller to execute the operation on the lockable resource if
12 the controller holds the lock on the lockable resource;
13 allowing the controller to acquire the lock if the controller does not hold
14 the lock on the lockable resource; and
15 allowing the controller to execute the operation on the lockable resource if
16 the controller acquires the lock.

1 12. (Once Amended) A computer-readable storage medium storing
2 instructions that when executed by a computer cause the computer to perform a
3 method for providing concurrency control for a policy-based management system
4 that controls resources in a distributed computing system, the method comprising:
5 receiving a request to perform an operation on a lockable resource from a
6 controller in the distributed computing system, wherein the lockable resource
7 presents one or more independent locks providing access to independent sub-units
8 of the resource;

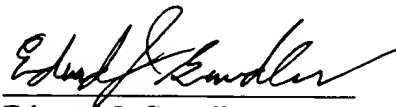
9 wherein the controller sends the request in order to enforce a first policy
10 for controlling resources in the distributed computing system;
11 determining whether the controller holds a lock on the lockable resource;
12 allowing the controller to execute the operation on the lockable resource if
13 the controller holds the lock on the lockable resource;
14 allowing the controller to acquire the lock if the controller does not hold
15 the lock on the lockable resource; and
16 allowing the controller to execute the operation on the lockable resource if
17 the controller acquires the lock.

1 16. (Once Amended) An apparatus that provides concurrency control
2 within a policy-based management system that controls resources in a distributed
3 computing system, the apparatus comprising:
4 a receiving mechanism that receives a request to perform an operation on a
5 lockable resource from a controller in the distributed computing system, wherein
6 the lockable resource presents one or more independent locks providing access to
7 independent sub-units of the resource;
8 wherein the controller sends the request in order to enforce a first policy
9 for controlling resources in the distributed computing system;
10 a determining mechanism that determines whether the controller holds a
11 lock on the lockable resource;
12 an execution mechanism that is configured to,
13 allow the controller to acquire the lock if the controller
14 does not hold the lock on the lockable resource, and to
15 allow the controller to execute the operation on the lockable
16 resource if the controller holds the lock on the lockable resource.

CONCLUSION

It is submitted that the present application is presently in form for allowance. Such action is respectfully requested.

Respectfully submitted,

By 
Edward J. Grundler
Registration No. 47,615

Date: July 18, 2002

Edward J. Grundler
PARK, VAUGHAN & FLEMING LLP
508 Second Street, Suite 201
Davis, CA 95616-4692
Tel: (530) 759-1663
FAX: (530) 759-1665